UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/043,496	01/10/2002	Binyu Zang	42390P13145	9184

8791 7590 06/23/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
CHOW, CHIH CHING	
ART UNIT	**PAPER NUMBER**
2192	

DATE MAILED: 06/23/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/043,496	ZANG ET AL.	
	Examiner	Art Unit	
	Chih-Ching Chow	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 February 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 February 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *04/15/02*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to amendment dated February 23, 2005.

2. Per Applicants' request, the Specification, Drawings (Fig. 2), Claims 1-4, 6-8, 11-14, 16-19, 21-24, and 26-28 have been amended (actually Claim 19 is the same as the original claim). Claims 1-30 remain pending.

Response to Amendment

3. Rejection Under 35 USC § 112

   - Applicants' amendment dated 02/23/2005, responding to the 11/19/2004 Office action provided in the 35 USC § 112 (2nd paragraph) rejection for Claims 1, 11, and 21. The Examiner has reviewed the amended Claims respectively, the 35 USC § 112 (2nd paragraph) rejection for Claims 1, 11, and 21 is hereby withdrawn.
   - Applicants' amendment dated 02/23/2005, responding to the 11/19/2004 Office action provided in the 35 USC § 112 (2nd paragraph) rejection for Claims 6-8, 16-18, and 26-28. The Examiner has reviewed the amended Claims respectively, the 35 USC § 112 (2nd paragraph) rejection for Claims 6-8, 16-18, and 26-28 is hereby withdrawn.

4. Rejection Under 35 USC § 103

   Applicants' amendment dated 02/23/2005, responding to the 11/19/2004 Office action provided in the 35 USC § 103 rejection for Claims 1-28. The Examiner has reviewed the amended Claims respectively. The Examiner is maintaining the 35 USC 103 rejections.

Response to Arguments

5. Applicants' arguments for Claims 1-28 have been fully considered respectfully by the examiner but they are not persuasive.

6. Applicants' arguments are basically in the following points:

- "Yeager does not disclose, either expressly or inherently: (i) determining a last use of a first canonical register in the block of code after a renaming, the first canonical register being mapped to the first original register; and (ii) applying one of a first rollback and a first recovery to the first original register based on whether the determined last use of the first canonical register occurs before a last definite write to the first original register in the block of code" - under REMARKS/ARGUMENTS, page 13, 2nd paragraph of the 02/23/2005 Amendment.

Examiner's Response: Even the amended claim adds new limitation to the claim, it's still covered in Yeager's disclosure. See Yeager's column 2, lines 40-46, "The present invention offers a highly efficient mechanism for saving and restoring register-renaming information to facilitate branch prediction and speculative execution. This mechanism enables the contents of register renaming mapping tables (i.e., memories used to rename logical register numbers to physical register numbers) to be saved and restored in a single clock cycle." - Yeager teaches a 'register rename mapping table', which keeps track of the use of registers (no matter it's first use or last use). Therefore, Yeager's disclosure inherently teaches 'determining' last use of the registers,

and applying rollback or recovery to the registers if last use of the register BEFORE a last write.

- "Yeager is directed to a superscalar processor architecture in which redundant mapping tables are used to rename registers and perform branch prediction. It is well known in the art that superscalar processor architectures operate on a different basis than a binary translation processor as is used in the present invention." -- under REMARKS/ARGUMENTS, page 13, 4th paragraph of the 02/23/2005 Amendment.

Examiner's Response: See MPEP 7.37.08 Unpersuasive Argument: Arguing Limitations Which Are Not Claimed

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., 'method for a binary translation processor') are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

- "Calaims 11 and 21 are directed to a computer program product and system, respectively, and which each share the same novel features of claim 1." -- under REMARKS/ARGUMENTS, page 14, 3rd paragraph of the 02/23/2005 Amendment.

Examiner's Response: Same comments as the previous item.

7. Examiner is maintaining the 35 USC 103 Rejections as set forth above. For the Applicants' convenience the rejections along with the amended claims are listed below.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 1, 11 and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,758,112 by Kenneth C. Yeager (hereinafter "Yeager"), in view of US 2004/0186981 by David S. Christie (hereinafter "Christie") .

CLAIM	**Yeager / Christie**
1. A method comprising: (a) determining a last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and (b) applying one of a first rollback and a first recovery to the first original register based on whether the determined last use of the first canonical register occurs before a last definite write to the first original register in the block of code.	For item (a), in Yeager, column 6, lines 25-34, "**Register renaming** is a technique used to keep track of changing register values. Processor 100 generates a myriad of temporary register results as it executes instructions. These temporary values, along with permanent values, are **stored** (*recording*) in register files 302 or 306 along with permanent values. Temporary values become new permanent values when the corresponding instructions graduate. Renaming identifies where the current (temporary or permanent)

value of each register is in the register file." Further, see Yeager's column 2, lines 40-46, "The present invention offers a highly efficient mechanism for saving and restoring register-renaming information to facilitate branch prediction and speculative execution. This mechanism enables the contents of register renaming mapping tables (i.e., memories used to rename logical register numbers to physical register numbers) to be saved and restored in a single clock cycle." - Yeager teaches a 'register rename mapping table', which keeps track of the use of registers (no matter it's first use or last use). Therefore, Yeager's disclosure inherently teaches 'determining' last use of the registers, and applying rollback or recovery to the registers if last use of the register BEFORE a last write. For item (b), in Yeager, column 7, lines 33-35, "mapping table 206 reflects the **latest** (*last definition*) mapping of logical destination register number 256. The **old** physical destination 282 associated with logical number 256 is output (*rollback*) from mapping table 206 and appended to active list (*recovery*)". Yeager teaches all aspects of claim 1 but does not mention the 'block of code' specifically. However, Christie teaches these features in an analogous art. In in Christie, paragraph 45, "in this particular implementation, MMU 20 generates the operating mode

responsive to a **code segment** descriptor corresponding to the **code being executed** (*block of code*) and further responsive to one or more values in control registers." And paragraph 78, " such a processor embodiment may execute interpreter software which reads each non-native instruction in a non-native code sequence as data, and executes **various software routines** (*block of code*) which emulate the defined operation of the non-native instruction as defined in the non-native processor architecture." In paragraph 40, "Alternatively, execution core 14 may employ a form of **register renaming** in which **any register within register file 22 may be mapped to an architected register**." Also, in paragraph, 0105, "FIG. 10 illustrates **mapping the general registers** 1052 **to registers in register file** 1044, any other non-native architected state may be mapped to registers in register file 1044. For example, **any of segment registers 1054, control registers** 1056, **or other registers** 1058 (or portions of any of these registers) may be mapped to register file 1044, as desired." Christie teaches the 'renaming of the register', but he is using the 'mapping' instead of the 'renaming'. Christie's 'general register', 'control register' or 'other register' basically function the same as the 'original register', 'temporary register' and the 'canonical

	register' as recited in current application. It would have been obvious to a person of ordinary skill in the art at the time of the invention was made to supplement Yeager's disclosure of the register renaming with utilizing it for 'block of code' concept taught by Christie, for the purpose of corresponding the code being executed to one or more values in control registers. (see Christie, paragraph 45, 1st sentence).
2. The method of claim 1 wherein applying one of the first rollback and the first recovery comprises: (a) applying the first rollback to the first original register if the determined last use of the first canonical register occurs before the last definite write to the first original register; and (b) applying the first recovery to the first original register if the determined last use of the first canonical register does not occur before the last definite write to the first original register.	For the feature of claim 1 see claim 1 rejection. In Yeager, Column 14, lines 33-35, "When an instruction graduates, the "old" (*determining the 'before' situation in item (b)*) physical register number associated with its destination register is written back (*rollback*) into the appropriate free list."
3. The method of claim 2 wherein applying the first rollback comprises: (a) replacing a first reference to a first target register with the first canonical register when the first reference is a destination of a last write to the first target register, the first target register corresponding to the first original register after the	For the feature of claim 2 see claim 2 rejection. For the rest of the feature in claim 3 see claim 1 rejection.

renaming; and
(b) replacing a second reference to the first target register with the first canonical register when the second reference is a source of a first operation after the last write to the first target register.

4. The method of claim 3 wherein applying the first recovery comprises:
copying the first target register to the first canonical register at end of the block.

For the feature of claim 2 see claim 2 rejection. See claim 1 rejection, where "temporary values become new permanent values" means the **copying** the target register to the first canonical register.

5. The method of claim 4 wherein copying the first target register comprises:
(a) copying the first target register to a first unused temporary register; and
(b) copying the first unused temporary register to the first canonical register.

For the feature of claim 4 see claim 4 rejection, for the rest of the feature of claim 5 see claim 4 rejection.

6. The method of claim 5 further comprises:
(a) recording a last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and
(b) applying one of a second rollback and a second recovery to the second original register based on whether the determined second last use of the second canonical register occurs before a last definite write to the second

For the feature of claim 5 see claim 5 rejection. For the rest of the features see claim 1 rejection.

original register in the block of code.

7. The method of claim 6 wherein applying one of the second rollback and the second recovery comprises: (a) applying the second rollback to the second original register if the determined last use of the second canonical register occurs before the last definite write to the second original register; and (b) applying the second recovery to the second original register if the determined last use of the second canonical register does not occur before the last definite write to the second original register.	For the feature of claim 6 see claim 6 rejection. For the rest of the features see claim 2 rejection.
8. The method of claim 7 wherein applying the second rollback comprises: (a) replacing a third reference to a second target register with the second canonical register when the third reference is a destination of a last write to the second target register, the second target register corresponding to the second original register after the renaming; and (b) replacing a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the last write to the second target register.	For the feature of claim 7 see claim 7 rejection. For the rest of the features see claim 3 rejection.
9. The method of claim 8 wherein	For the feature of claim 8 see claim 8

applying the second recovery comprises: (a) copying the second target register to the second canonical register at end of the block.	rejection. For the rest of the features see claim 4 rejection.
10. The method of claim 9 wherein copying the second target register comprises: (a) copying the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and (b) copying the second unused temporary register to the second canonical register.	For the feature of claim 9 see claim 9 rejection. For rest of the features see claim 5 rejection.
11. A computer program product comprising: (a) a machine useable medium having program code embedded therein, the program code comprising: (i) computer readable program code to determine a last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and (ii) computer readable program code to apply one of a first rollback and a first recovery to the first original register based on whether the determined last use of the first canonical register occurs before a last definite write to the first original	Both Yeager's FIG. 1 (Instruction Cache) and Christie's Fig. 13 (L2 Cache) teaches a machine useable medium having code embedded therein. For the rest of feature 11, same as claim 1 rejection.

register in the block of code.

12. The computer program product of claim 11 wherein the computer readable program code to apply one of the first rollback and the first recovery comprises:
(a) computer readable program code to apply the first rollback to the first original register if the determined last use of the first canonical register occurs before the last definite write to the first original register; and
(b) computer readable program code to apply the first recovery to the first original register if the determined last use of the first canonical register does not occur before the last definite write to the first original register.

For the feature of claim 11 see claim 11 rejection. For rest of the features see claim 2 rejection.

13. The computer program product of claim 12 wherein the computer readable program code to apply the first rollback comprises:
(a) computer readable program code to replace a first reference to a first target register with the first canonical register when the first reference is a destination of a last write to the first target register, the first target register corresponding to the first original register after the renaming; and
(b) computer readable program code to replace a second reference to the first target register with the first canonical register when the second reference is a

For the feature of claim 12 see claim 12 rejection. For rest of the features see claim 3 rejection.

source of a first operation after the last write to the first target register.

14. The computer program product of claim 13 wherein the computer readable program code to apply the first recovery comprises:
computer readable program code to copy the first target register to the first canonical register at end of the block.

For the feature of claim 12 see claim 12 rejection. For rest of the features see claim 4 rejection.

15. The computer program product of claim 14 wherein the computer readable program code to copy the first target register comprises:
(a) computer readable program code to copy the first target register to a first unused temporary register; and
(b) computer readable program code to copy the first unused temporary register to the first canonical register.

For the feature of claim 14 see claim 14 rejection. For rest of the features see claim 5 rejection.

16. The computer program product of claim 15 further comprises:
(a) computer readable program code to record a last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and
(b) computer readable program code to apply one of a second rollback and a second recovery to the second original register based on whether the determined last use of the second

For the feature of claim 15 see claim 15 rejection. For rest of the features see claim 6 rejection.

canonical register occurs before a last definite write to the second original register in the block of code.

17. The computer program product of claim 16 wherein the computer readable program code to apply one of the second rollback and the second recovery comprises:

(a) computer readable program code to apply the second rollback to the second original register if the determined last use of the second canonical register occurs before the last definite write to the second original register; and

(b) computer readable program code to apply the second recovery to the second original register if the determined last use of the second canonical register does not occur before the last definite write to the second original register.

For the feature of claim 16 see claim 16 rejection. For rest of the features see claim 7 rejection.

18. The computer program product of claim 17 wherein the computer readable program code to apply the second rollback comprises:

(a) computer readable program code to replace a third reference to a second target register with the second canonical register when the third reference is a destination of a last write to the second target register, the second target register corresponding to the second original register after the renaming; and

(b) computer readable program code to

For the feature of claim 17 see claim 17 rejection. For rest of the features see claim 8 rejection.

replace a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the last write to the second target register.

19. The computer program product of claim 18 wherein the computer readable program code to apply the second recovery comprises:
computer readable program code to copy the second target register to the second canonical register at end of the block.

For the feature of claim 18 see claim 18 rejection. For rest of the features see claim 9 rejection.

20. The computer program product of claim 19 wherein the computer readable program code to copy the second target register comprises:
(a) computer readable program code to copy the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and
(b) computer readable program code to copy the second unused temporary register to the second canonical register.

For the feature of claim 19 see claim 19 rejection. For rest of the features see claim 10 rejection.

21. A system comprising:
(a) a processor; and
(b) a memory coupled to the processor to store program code, the program code, when executed, causing the

Both Yeager's FIG. 1 (Superscalar Processor Architecture) and Christie's Fig. 13 (Processor) teaches a system with a processor and memory to store program code. For the rest of feature

processor to: (i) determine a last use of a first canonical register in a block of code after a renaming, the first canonical register being mapped to a first original register; and (ii) apply one of a first rollback and a first recovery to the first original register based on whether the determined last use of the first canonical register occurs before a last definite write to the first original register in the block of code.	21, same as claim 1 rejection.
22. The system of claim 21 wherein the program code causing the processor to apply one of the first rollback and the first recovery causes the processor to: (a) apply the first rollback to the first original register if the determined last use of the first canonical register occurs before the last definite write to the first original register; and (b) apply the first recovery to the first original register if the determined last use of the first canonical register does not occur before the last definite write to the first original register.	For the feature of claim 21 see claim 21 rejection. For rest of the features see claim 2 rejection.
23. The system of claim 22 wherein the program code causing the processor to apply the first rollback causes the processor to: (a) replace a first reference to a first target register with the first canonical register when the first reference is a	For the feature of claim 22 see claim 22 rejection. For rest of the features see claim 3 rejection.

destination of a last write to the first target register, the first target register corresponding to the first original register after the renaming; and

(b) replace a second reference to the first target register with the first canonical register when the second reference is a source of a first operation after the last write to the first target register.

24. The system of claim 23 wherein the program code causing the processor to apply the first recovery causes the processor to:

(a) copy the first target register to the first canonical register at end of the block.

For the feature of claim 22 see claim 22 rejection. For rest of the features see claim 4 rejection.

25. The system of claim 24 wherein the program code causing the processor to copy the first target register causes the processor to:

(a) copy the first target register to a first unused temporary register; and

(b) copy the first unused temporary register to the first canonical register.

For the feature of claim 24 see claim 24 rejection. For rest of the features see claim 5 rejection.

26. The system of claim 25 wherein the program code further causes the Processor to:

(a) determine a last use of a second canonical register in the block of code after the renaming, the second canonical register being mapped to a second original register; and

For the feature of claim 25 see claim 25 rejection. For rest of the features see claim 6 rejection.

(b) apply one of a second rollback and a second recovery to the second original register based on whether the determined last use of the second canonical register occurs before a last definite write to the second original register in the block of code.

27. The system of claim 26 wherein the program code causing the processor to apply one of the second rollback and the second recovery causes the processor to:

(a) apply the second rollback to the second original register if the determined last use of the second canonical register occurs before the last definite write to the second original register; and

(b) apply the second recovery to the second original register if the determined last use of the second canonical regiester does not occur before the last definite write to the second original register.

For the feature of claim 26 see claim 26 rejection. For rest of the features see claim 7 rejection.

28. The system of claim 27 wherein the program code causing the processor to apply the second rollback causes the processor to:

(a) replace a third reference to a second target register with the second canonical register when the third reference is a destination of a last write to the second target register, the second target register corresponding to

For the feature of claim 27 see claim 27 rejection. For rest of the features see claim 8 rejection.

the second original register after the renaming; and
(b) replace a fourth reference to the second target register with the second canonical register when the fourth reference is a source of a second operation after the last write to the second target register.

29. The system of claim 28 wherein the program code causing the processor to apply the second recovery causes the processor to:
(a) copy the second target register to the second canonical register at end of the block.

For the feature of claim 28 see claim 28 rejection. For rest of the features see claim 8 rejection.

30. The system of claim 29 wherein the program code causing the processor to copy the second target register causes the processor to:
(a) copy the second target register to a second unused temporary register before copying the first unused temporary register to the first canonical register; and
(b) copy the second unused temporary register to the second canonical register.

For the feature of claim 29 see claim 20 rejection. For rest of the features see claim 10 rejection.

Conclusion

10. The following summarizes the status of the claims:

35 USC § 103 rejection: Claims 1-30

11. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL.** See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chih-Ching Chow whose telephone number is 571-272-3693. The examiner can normally be reached on 7:30am - 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Any inquiry of a general nature of relating to the status of this application should be directed to the **TC2100 Group receptionist: 571-272-2100.**

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Chih-Ching Chow

Examiner

Art Unit 2192

June 08, 2005

CC

TUAN DAM
SUPERVISORY PATENT EXAMINER